

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-168011

(43)Date of publication of application : 14.06.1994

(51)Int.Cl.

G05B 19/05

G06F 9/06

(21)Application number : 04-319178

(71)Applicant : TOSHIBA ENG CO LTD
TOSHIBA CORP

(22)Date of filing : 30.11.1992

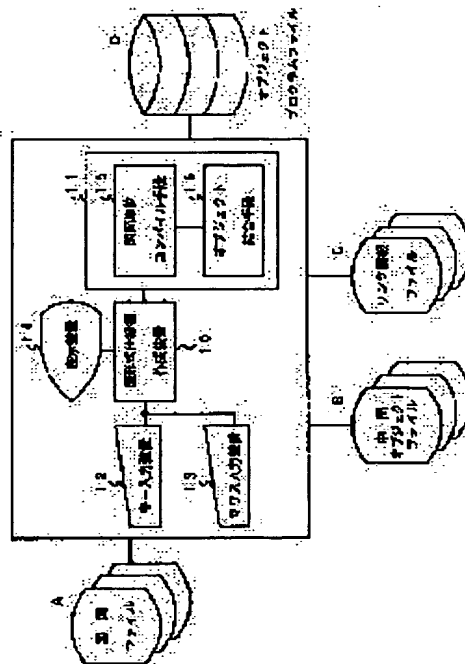
(72)Inventor : ENDO TOSHIHIRO
ITAGAKI KOICHI

(54) PROGRAM GENERATING DEVICE

(57)Abstract:

PURPOSE: To improve program generation efficiency at the time of correcting a program by compiling graphic specifications by the unit of graphic specification, generating an intermediate object file in the unit of graphic specification and generating a link information file consisting of connection information between graphic specifications.

CONSTITUTION: A program generating device is provided with a graphic specification generating device 10 and a compiling device 11 compiling the specification and generating an object program. The drawing unit compiling means 15 of the compiling device 11 generates an intermediate object file B in a diagram unit and a link information file C. An object connection means 16 connects the respective object files B based on the link information file C and generates the object program by connecting the respective intermediate object files B based on the link information file C. Thus, the graphic specification of the programs for plural pieces can be compiled in the respective graphic specification unit, and program generation efficiency at the time of correcting the program can considerably be improved.



LEGAL STATUS

[Date of request for examination] 26.03.1998

[Date of sending the examiner's decision of rejection] 11.07.2000

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平6-168011

(43)公開日 平成 6年(1994) 6月14日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 5 B 19/05		A 7361-3H		
G 0 6 F 9/06	4 3 0	D 9367-5B		

審査請求 未請求 請求項の数 1 (全 9 頁)

(21)出願番号 特願平4-319178

(22)出願日 平成 4年(1992)11月30日

(71)出願人 000221018

東芝エンジニアリング株式会社
神奈川県川崎市幸区堀川町66番 2

(71)出願人 000003078

株式会社東芝
神奈川県川崎市幸区堀川町72番地

(72)発明者 遠藤 利宏

神奈川県川崎市幸区堀川町66番 2 東芝エ
ンジニアリング株式会社内

(72)発明者 板垣 孝一

東京都府中市東芝町 1 番地 株式会社東芝
府中工場内

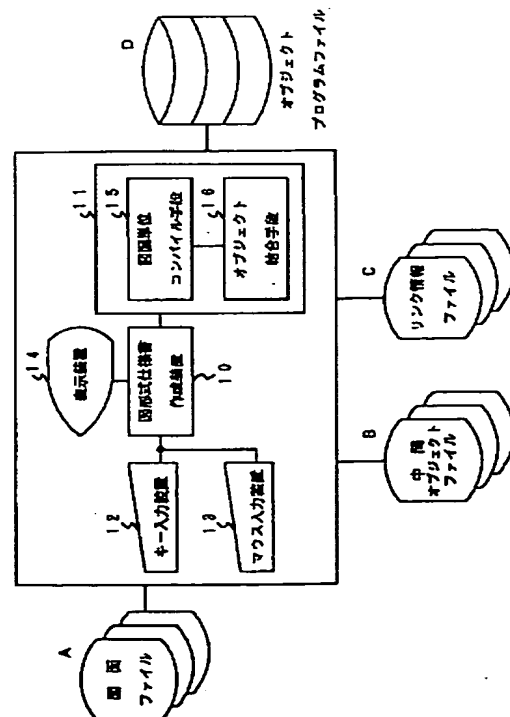
(74)代理人 弁理士 鈴江 武彦

(54)【発明の名称】 プログラム作成装置

(57)【要約】

【目的】本発明は、プログラム修正時のプログラム作成効率の改善を図る。

【構成】本発明のプログラム作成装置は、プログラムを表現した複数の図形式仕様書 F を各図形式仕様書毎にコンパイルし、そのコンパイルの過程で得られた情報から図形式仕様書単位の間接オブジェクトファイル B と各図形式仕様書間の接続情報からなるリンク情報ファイル C とを作成するコンパイル手段 1 5 と、前記リンク情報ファイル C の接続情報を用いて前記図形式仕様書単位の各中間オブジェクトファイル B を結合しオブジェクトプログラムを生成するオブジェクト結合手段 1 6 とを具備した構成である。



【特許請求の範囲】

【請求項1】 作成対象のプログラムを図形を用いて表現した図形式仕様書を入力してオブジェクトプログラムを作成するプログラム作成装置において、

所定のプログラムを表現した複数の図形式仕様書を各図形式仕様書毎にコンパイルし、そのコンパイルの過程で得られた情報から図形式仕様書単位の間接オブジェクトファイルと各図形式仕様書間の接続情報からなるリンク情報ファイルとを作成するコンパイル手段と、

前記リンク情報ファイルの接続情報を用いて前記図形式仕様書単位の各中間オブジェクトファイルを結合し、前記オブジェクトプログラムを生成するオブジェクト結合手段と、を具備したことを特徴とするプログラム作成装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、プロセス制御用プログラマブルコントローラのプログラムを作成するために用いることのできるプログラム作成装置に係り、さらに詳しくは図形式仕様書をコンパイルしてオブジェクトプログラムを作成するプログラム作成装置に関する。

【0002】

【従来の技術】 近年、デジタル計装の分野では、コントローラの制御プログラムを図形で表現した図形式仕様書を作り、その図形式仕様書をコンパイルしてオブジェクトプログラムを自動生成するプログラム作成装置が開発されている。そして、制御の高度化に伴い、1つのプロセス制御を実現するためには多くの図形式仕様書が作成されるようになってきた。

【0003】 かかるプログラム作成装置は、あるプログラムを作成するためにはプログラムを表現した一連の図形式仕様書の全て解析して、図形式仕様書に対応する1つのオブジェクトプログラムを作成する。そして制御内容の段階的な進歩に合わせてプロセス制御の内容も頻繁に修正が加えられるようになってきた。

【0004】 ところが、プログラム修正時には、修正対象となるプログラムに関する一連の図形式仕様書を全て再コンパイルする必要がある。そのため、1つのプログラムに係る図形式仕様書の数が増大した現在では、プログラム修正の都度その膨大な図形式仕様書を再コンパイルしていたためプログラム作成効率が著しく低下することになる。

【0005】

【発明が解決しようとする課題】 この様に、従来のプログラム作成装置は、頻繁に行われるプログラム修正の度に膨大な数の図形式仕様書を再コンパイルする必要があったため、プログラム作成効率が低下する問題があった。

【0006】 本発明は以上のような実情に鑑みてなされたもので、複数枚に亘るプログラムの図形式仕様書を各

図形式仕様書単位でコンパイルでき、プログラム修正時のプログラム作成効率を大幅に改善できるプログラム作成装置を提供することを目的とする。

【0007】

【課題を解決するための手段】 上記目的を達成するために本発明のプログラム作成装置は、所定のプログラムを表現した複数の図形式仕様書を各図形式仕様書毎にコンパイルし、そのコンパイルの過程で得られた情報から図形式仕様書単位の間接オブジェクトファイルと各図形式仕様書間の接続情報からなる仕様書接続情報ファイルとを作成するコンパイル手段と、前記仕様書接続情報ファイルの接続情報を用いて前記図形式仕様書単位の各中間オブジェクトファイルを結合し、前記オブジェクトプログラムを生成するオブジェクト結合手段とを具備して構成される。

【0008】

【作用】 以上のように構成された本発明のプログラム作成装置では、所定のプログラムを表現した複数の図形式仕様書が各図形式仕様書毎にコンパイル手段によりコンパイルされて、図形式仕様書単位の間接オブジェクトファイルと各図形式仕様書間の接続情報からなるリンク情報ファイルとが作成される。そしてオブジェクト結合手段により図形式仕様書単位の各中間オブジェクトファイルがリンク情報ファイルの接続情報を用いて結合されてオブジェクトプログラムが生成される。

【0009】 この様に、図形式仕様書を図形式仕様書単位でコンパイルし、かつ中間オブジェクトファイルを図形式仕様書単位で作成すると共に図形式仕様書間の接続情報からなるリンク情報ファイルを作成するため、プログラム修正時には修正が加えられる一部の図形式仕様書のみをコンパイルすればよいことになる。

【0010】

【実施例】 以下、本発明の実施例について説明する。

【0011】 図1には本発明の一実施例に係るプログラム作成装置の機能ブロックが示されている。本実施例のプログラム作成装置は、例えばプロセス制御用プログラマブルコントローラのプログラムを図形を用いて表現した図形式仕様書を作成する図形式仕様書作成装置10と、この図形式仕様書作成装置10から入力する図形式仕様書をコンパイルしてオブジェクトプログラムを作成するコンパイル装置11とを備えている。

【0012】 図形式仕様書作成装置10は、キーボード及びマウスを用いた入力装置12、13、及び表示装置14が接続されている。入力装置12、13からは図形式仕様書の作成に必要な図形データ及びテキストデータが入力され、作成中の図形式仕様書が表示装置14に表示される。

【0013】 コンパイル装置11は、図面単位の間接オブジェクトファイルBと図面間のリンク情報ファイルCとを作成する図面単位コンパイル手段15と、リンク情

(11) 報ファイルに基づいて各中間オブジェクトファイルを結合しオブジェクトプログラムを生成するオブジェクト結合手段16とを備えている。図2にコンパイル装置11の具体的な構成を示す。

【0014】図面ファイルA1～A3には、図形式仕様書作成装置10によりあるプロセス制御に関連して作成された一連の図形式仕様書が格納される。図面単位コンパイル手段15は、複数の図面ファイルA1～A3から任意の図面ファイルを読込んで1の図形式仕様書内のプログラムオブジェクト情報と、他の図形式仕様書との接続情報であるリンク情報とをそれぞれ抽出し、当該図面ファイルに対応させて中間オブジェクトファイルB及びリンク情報ファイルCを作成する。すなわち、図面ファイルA1をコンパイルすることにより得られるプログラムオブジェクト情報から当該図面ファイルA1の中間オブジェクトファイルB1を作成し、同図面ファイルのコンパイルで得られる図形式仕様書間の接続情報からリンク情報ファイルC1を作成する。同様に、図面ファイルA2の時は中間オブジェクトファイルB2、リンク情報ファイルC2を作成し、図面ファイルA3の時は中間オブジェクトファイルB3、リンク情報ファイルC3を作成する。

【0015】ここで、プログラムオブジェクト情報とは、コンパイル過程の構文解析等で得られる情報であり、図形式仕様書内のシンボル、シンボルの接続状況、プロセス制御用の信号名称等からなる。このプログラムオブジェクト情報にリンク情報は含まない。またリンク情報とは、同様にコンパイル過程の構文解析等で得られる情報であり、一連の図形式仕様書間の接続関係の情報である。オブジェクト結合手段16は、後述する処理によって作成するオブジェクトプログラムをオブジェクトプログラムファイルDに格納する。

【0016】次に、以上のように構成された本実施例の動作について説明する。まず、図形式仕様書作成装置10において、あるプロセス制御を実現するためにプロセスを図形で表現した一連の図形式仕様書が作成される。図6には、プロセス制御用プログラマブルコントローラのシーケンスフローチャート機能を表した図形式仕様書F1～F3の例が示されている。各図形式仕様書F1～F3には図形式使用書間の接続情報を表す図形としてF4が使用されている。すなわち、図形式仕様書F1～F3は別々にコンパイル装置11に読込まれるが、F4に基づいてF1～F3の順に処理が実施されることがわかる。この図形式仕様書F1～F3が図面ファイルA1～A3に格納される。

【0017】次に、図面単位コンパイル手段15によって任意の図形式仕様書がコンパイルされる。図面単位コンパイル手段15は、図3に示すフローチャートに従って動作する。

【0018】例えば、図面ファイルA2がコンパイル指

定されると(ステップX1)、指定ファイルF2のコンパイルを実施する(ステップX2)。コンパイルの過程でコンパイルエラーのチェックを行い(ステップX3)、エラーが発生しなければ中間オブジェクトファイルB2およびリンク情報ファイルC2を作成する(ステップX4)。

【0019】図4は、図形式仕様書F2をコンパイルした際に生成されるリンク情報ファイルC2の内容を示している。同図に示すように、各図形F2が表す具体的な接続情報が格納されている。

【0020】また図形式仕様書F2内のシンボル、シンボルの接続状況、プロセス制御用の信号名称からなるプログラムオブジェクト情報が、中間オブジェクトファイルB2に格納される。

【0021】次に、他にコンパイル指定された図面ファイルが存在するか否かを判断し(ステップX5)、存在すれば再びステップX1に戻って同様の処理を実行し、存在しなければコンパイル処理を終了する(ステップX7)。なお、ステップX3においてコンパイルエラーが発生していればエラー処理を行う(ステップX6)。

【0022】一方、オブジェクト結合手段16は、図5に示すフローチャートに従って動作する。すなわち、中間オブジェクトファイルBが存在するか否かを判断し(ステップY1)、中間オブジェクトファイルBが有ればオブジェクト結合手段16によるオブジェクトプログラムの作成処理が実施される(ステップY2)。存在する全ての中間オブジェクトファイルBを、リンク情報ファイルCに格納された各中間オブジェクトファイルBの接続情報に基づいて結合して、例えばプロセス制御用プログラマブルコントローラのオブジェクトプログラムを作成する。プログラム作成の時にエラーが発生すれば(ステップY3)、エラー処理を実施する(ステップY5)。この様にして作成されたオブジェクトプログラムがオブジェクトプログラムファイルDに格納される。

【0023】またプロセス制御用プログラマブルコントローラのプログラムをバージョンアップする場合は、そのバージョンアップに伴い修正された一部の図形式仕様書のみを図面単位でコンパイルする。本実施例では図面単位(図形式仕様書単位)で中間オブジェクトファイルBを作成し、かつ図面間(図形式仕様書間)の接続情報であるリンク情報ファイルCを作成しているため、修正に係る図形式仕様書の中間オブジェクトファイルB及びリンク情報ファイルCのみを書き替えれば、バージョンアップされたオブジェクトプログラムを容易に作成できるものとなる。この様に本実施例によれば、図面単位の間オブジェクトファイルBを作成し、かつ図面間の接続情報であるリンク情報ファイルCを作成して、両ファイルを用いてオブジェクトプログラムを作成するように構成したので、プログラム修正時には修正に係る図形式仕様書のみをコンパイルするだけでオブジェクトプログ

ラムを作成でき、プログラム作成効率を飛躍的に向上させることができる。本発明は上記実施例に限定されるものではなく、本発明の要旨を逸脱しない範囲内で種々変形実施可能である。

【0024】

【発明の効果】以上詳記したように本発明によれば、複数枚に亘るプログラムの図形式仕様書を各図形式仕様書単位でコンパイルでき、プログラム修正時のプログラム作成効率を大幅に改善できるプログラム作成装置を提供できる。

【図面の簡単な説明】

【図1】本発明の一実施例に係るプログラム作成装置の機能ブロック図である。

【図2】図1に示すコンパイル装置の機能ブロック図である。

【図3】上記実施例のプログラム作成装置における図面

単位のコンパイル処理を示すフローチャートである。

【図4】リンク情報ファイルの具体例を示す図である。

【図5】上記実施例のプログラム作成装置におけるファイル結合処理を示すフローチャートである。

【図6】図形式仕様書の具体例を示す図である。

【符号の説明】

10…図形式仕様書作成装置
コンパイル手段

11…コンパイル装置
クト結合手段

12…キー入力装置
イル

13…マウス入力装置
ジェクトファイル

14…表示装置
報ファイル

15…図面単位

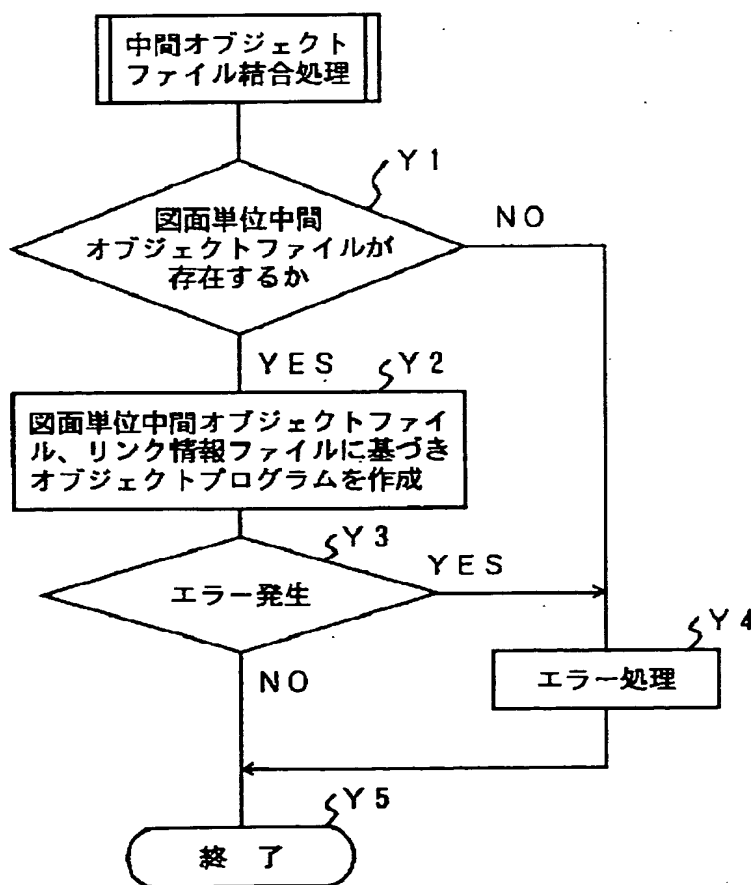
16…オブジェ

A…図面ファ

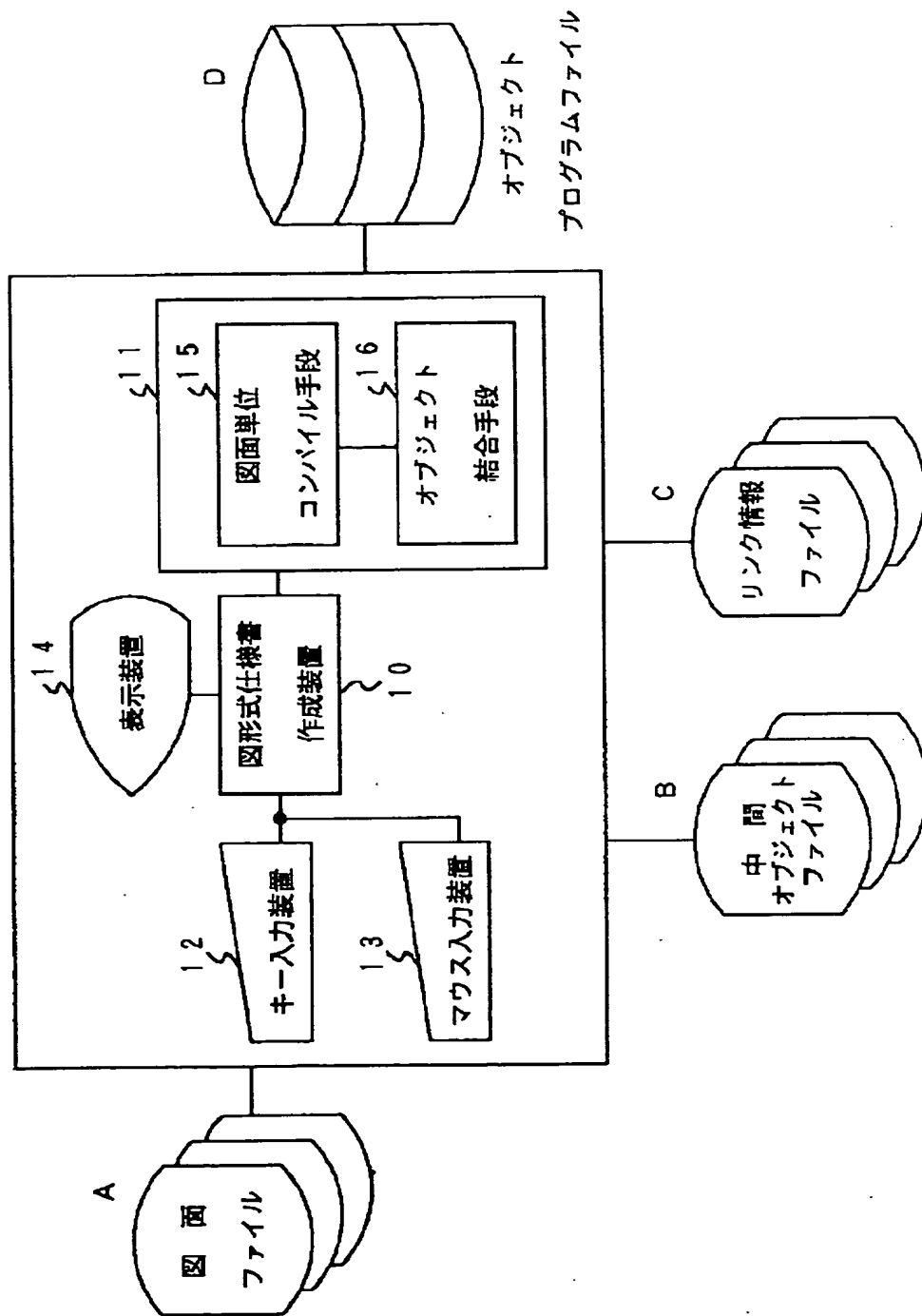
B…中間オブ

C…リンク情

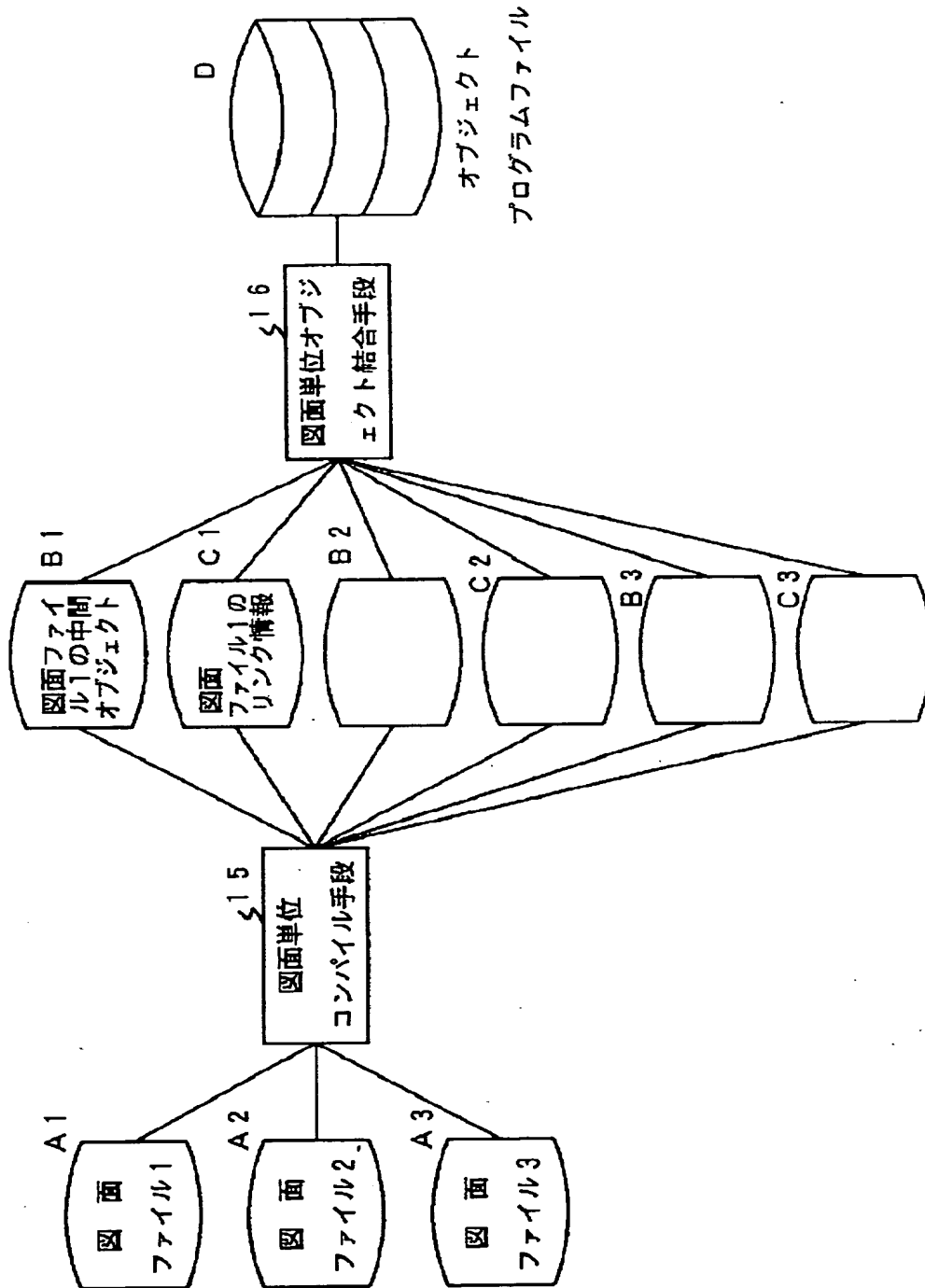
【図5】



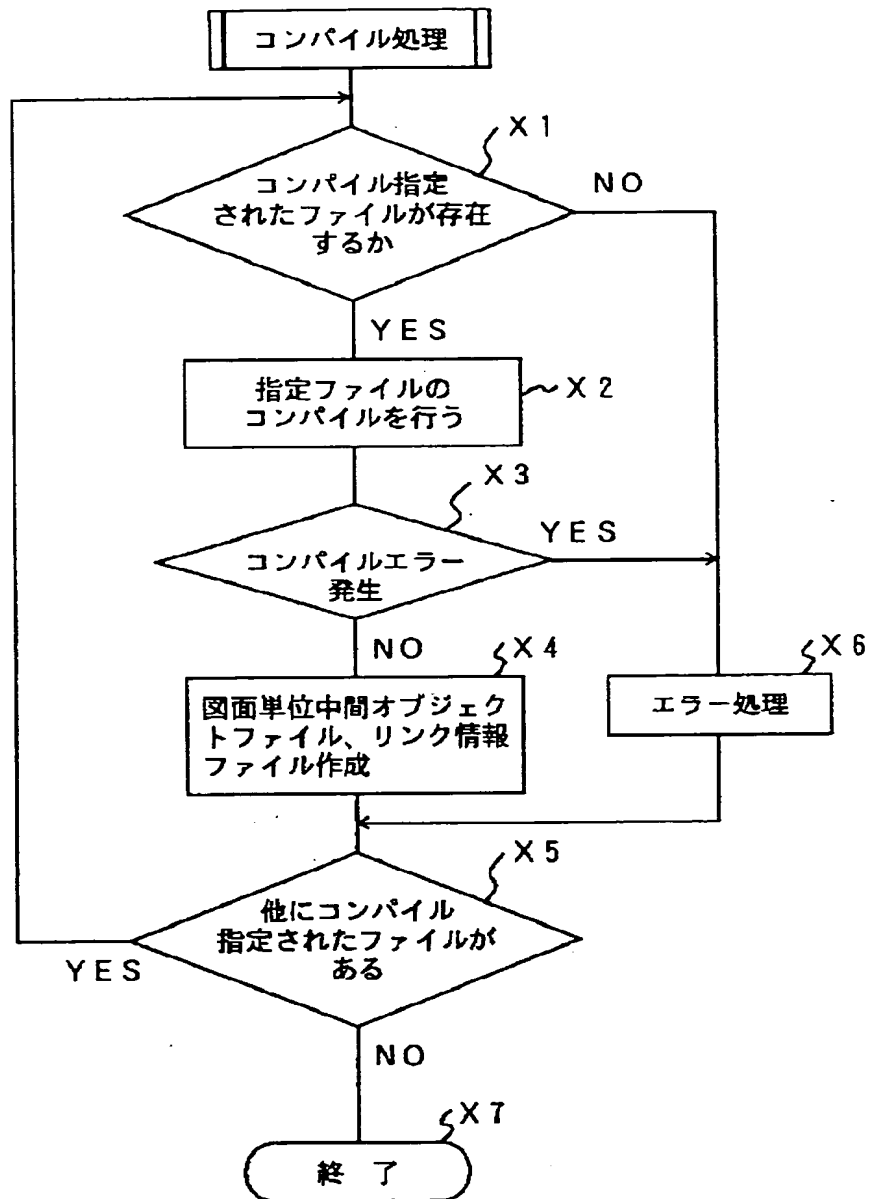
【図1】



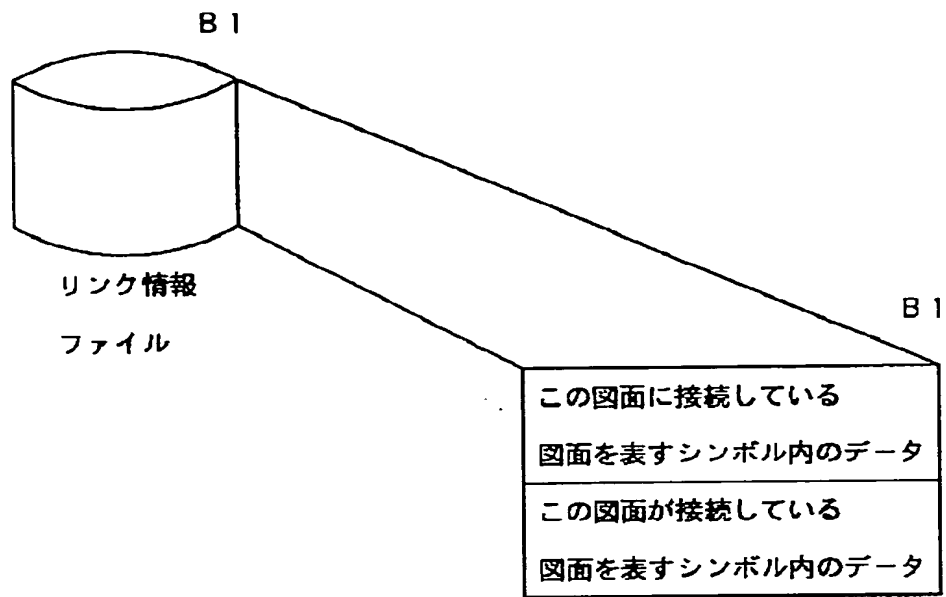
【図2】



【図3】



【図4】



【図6】

